***ЛАБОРАТОРНА РОБОТА №1***

**Тема:** Вивчення базових понять класу.

**Мета:** Закріпити базові знання про клас. Навчитись створювати класи засобами мови С++

***ПОРЯДОК ВИКОНАННЯ РОБОТИ***

**Завдання 1.** Створіть клас Int, що імітує стандартний тип int. Єдине поле цього класу повинно мати тип int. Створіть методи, які будуть встановлювати значення поля рівним нулю, ініціалізувати його цілим значенням, виводити значення поля на екран і складати два значення типу Int.

Напишіть програму, в якій будуть створені три об'єкти класу Int, два з яких будуть ініціалізованими. Додайте два ініціалізованних об'єкта, надайте результат третьому, а потім відобразіть результат на екрані.

Код програми

#include "stdafx.h"

#include<iostream>

using namespace std;

class Int {

private:

int Inte;

public:

void setInt(int \_Int) {

Inte = \_Int;

}

int getInt() {

return Inte;

}

void set0() {

Inte = 0;

}

int sum(int num1, int num2)

{

return num1 + num2;

}

};

int main()

{

Int int1, int2, int3;

int1.set0();

int2.setInt(2);

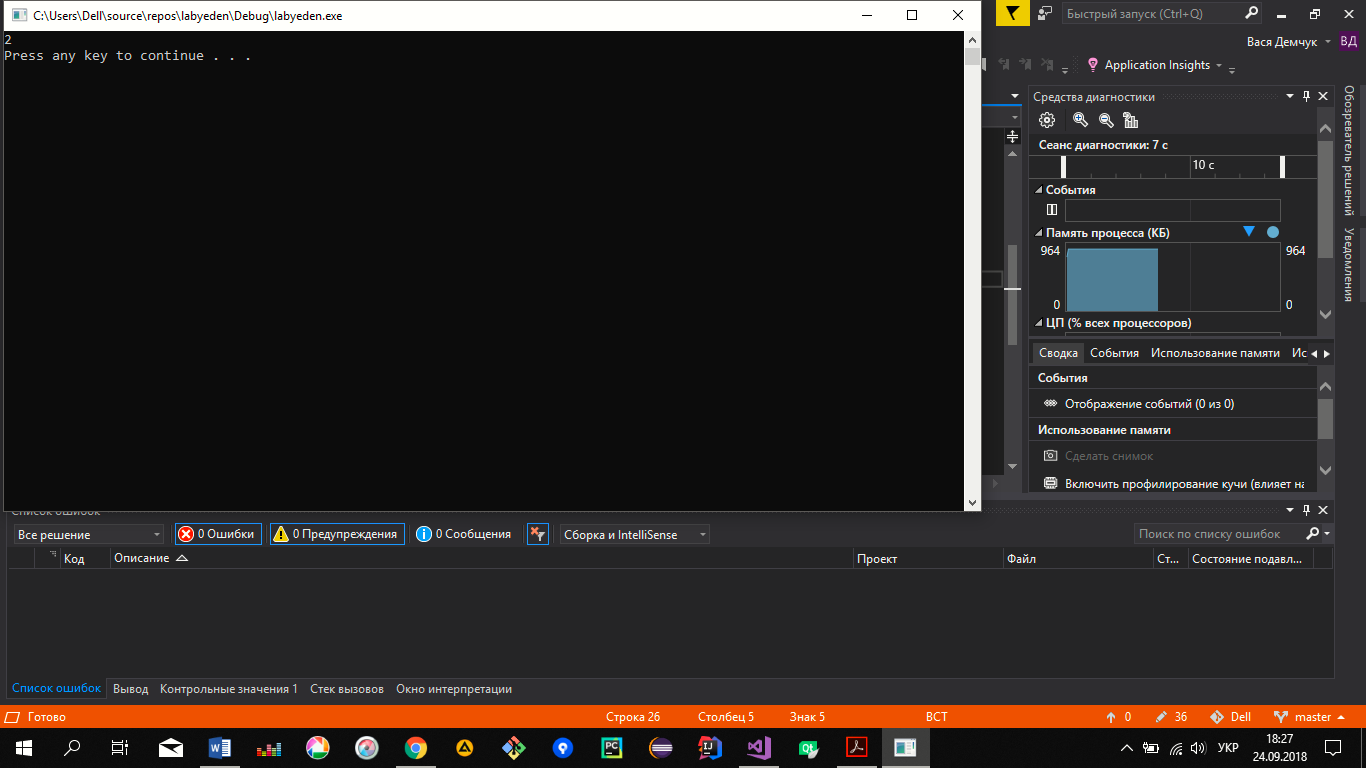
cout << int1.sum(int1.getInt(), int2.getInt())<<endl;

system("pause");

return 0;

}

Вивід програми



**Завдання 2.** Уявіть пункт для прийому платежів за проїзд по автостраді. Кожна проїжджаюча машина повинна заплатити за проїзд 50 центів, однак частина машин платить за проїзд, а частина проїжджає безкоштовно. У касі ведеться облік числа проїхавших машин і сумарна виручка від плати за проїзд.

Створіть модель такої каси за допомогою класу Kasa. Клас повинен містити два поля. Одне з них, типу unsigned int, призначене для обліку кількості проїхали автомобілів, а друге, що матиме тип double, міститиме сумарну виручку від оплати проїзду. Конструктор повинен ініціалізувати обидва поля нульовими значеннями. Метод payingCar () інкрементує число машин і збільшує на 0,50 сумарну виручку. Інший метод, nорауСаг (), збільшує на одиницю число автомобілів, але залишає без зміни виручку. Метод display () виводить обидва значення на екран. Там, де це можливо, зробіть методи константними.

Створіть програму, яка продемонструє роботу класу. Програма повинна запропонувати користувачеві натиснути одну клавішу для того, щоб зімітувати оплату водієм, і іншу клавішу, щоб зімітувати несумлінного водія. Натискання клавіші Esc повинно привести до видачі поточних значень кількості машин і виручки, і до завершення програми.

Код програми

#include "stdafx.h"

#include<iostream>

#include<conio.h>

using namespace std;

class Kasa {

private:

unsigned int numberCar;

double money;

public:

Kasa(int \_numberCar, double \_money) {

numberCar = \_numberCar;

money = \_money;

}

void payingCar() {

numberCar++;

money += 0.5;

}

void nopayingCar() {

numberCar++;

}

void display(){

cout << numberCar << " -"<<money<<"$"<<endl;

}

};

int main()

{

Kasa kasa(0, 0);

int c;

while (1){

c = \_getch();в

if (c == ' ')

{

kasa.nopayingCar();

}

if (c == 'e' || c == 'E')

{

kasa.payingCar();

}

if (c == 'q' || c == 'Q')

break;

kasa.display();

}

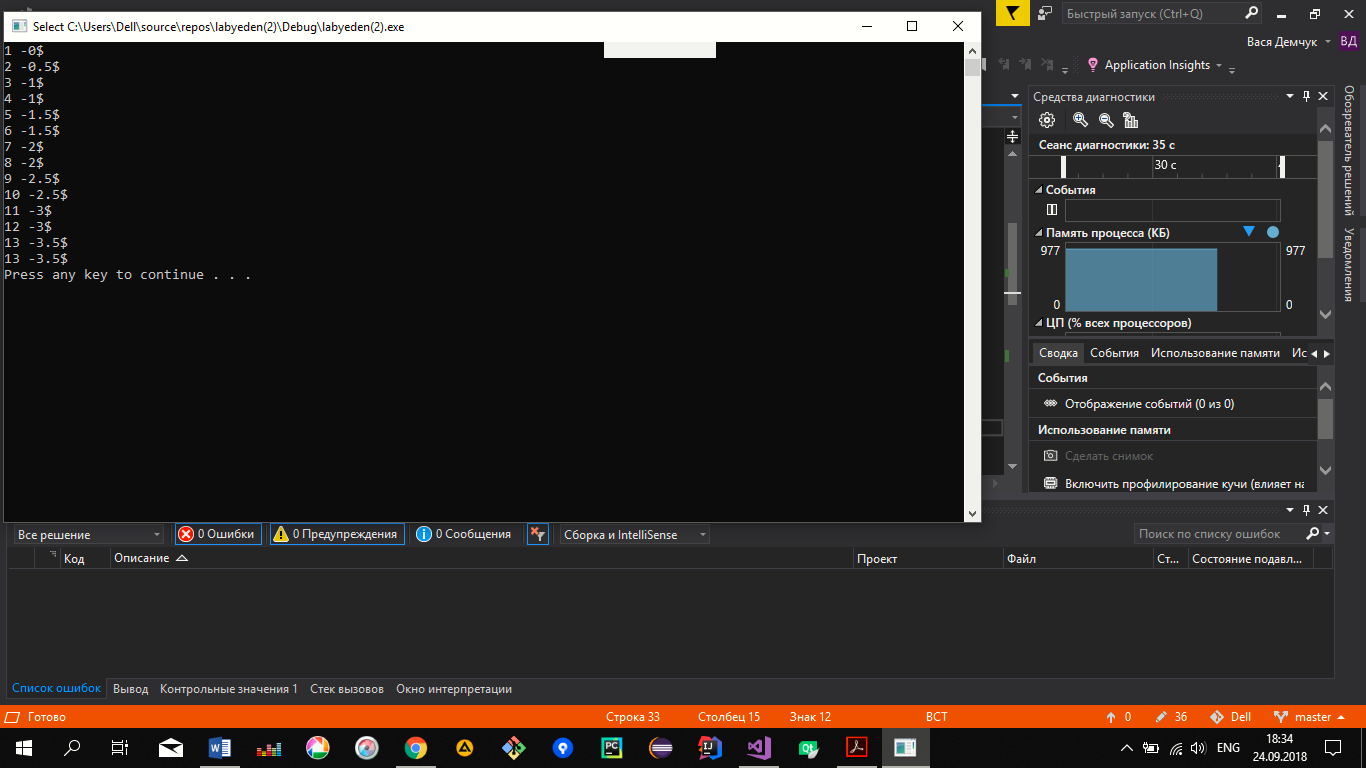
kasa.display();

system("pause");

return 0;

}

Вивід програми



**Завдання 3.** Створіть клас з ім'ям time, що містить три поля типу int, призначених для зберігання годин, хвилин і секунд. Один з конструкторів класу повинен ініціалізувати поля нульовими значеннями, а інший конструктор - заданим набором значень. Створіть

метод класу, який буде виводити значення полів на екран у форматі 11:59:59, і метод, складає значення двох об'єктів типу time, переданих в якості аргументів.

У функції main () слід створити два ініціалізованих об'єкта (подумайте, чи повинні вони бути константними) і один неініціалізованний об'єкт. Потім складіть два ініціалізованих значення, а результат надайте третьому об'єкту і виведіть його значення на екран. Де можливо, зробіть методи константними.

Код програми

#include "stdafx.h"

#include<iostream>

using namespace std;

class Time {

private:

int hour;

int minute;

int second;

public:

Time(){}

Time(int \_hour, int \_minute, int \_second)

{

hour = \_hour;

minute = \_minute;

second = \_second;

}

void getTime() {

cout << hour << ":" << minute << ":" << second << endl;

}

void sumTime(Time time1, Time time2)

{

hour = time1.hour + time2.hour;

minute = time1.minute + time2.minute;

second = time1.second + time2.second;

}

};

int main()

{

Time time1(0,0,0), time2(12,32,13);

Time time3;

time3.sumTime(time1, time2);

time1.getTime();

time2.getTime();

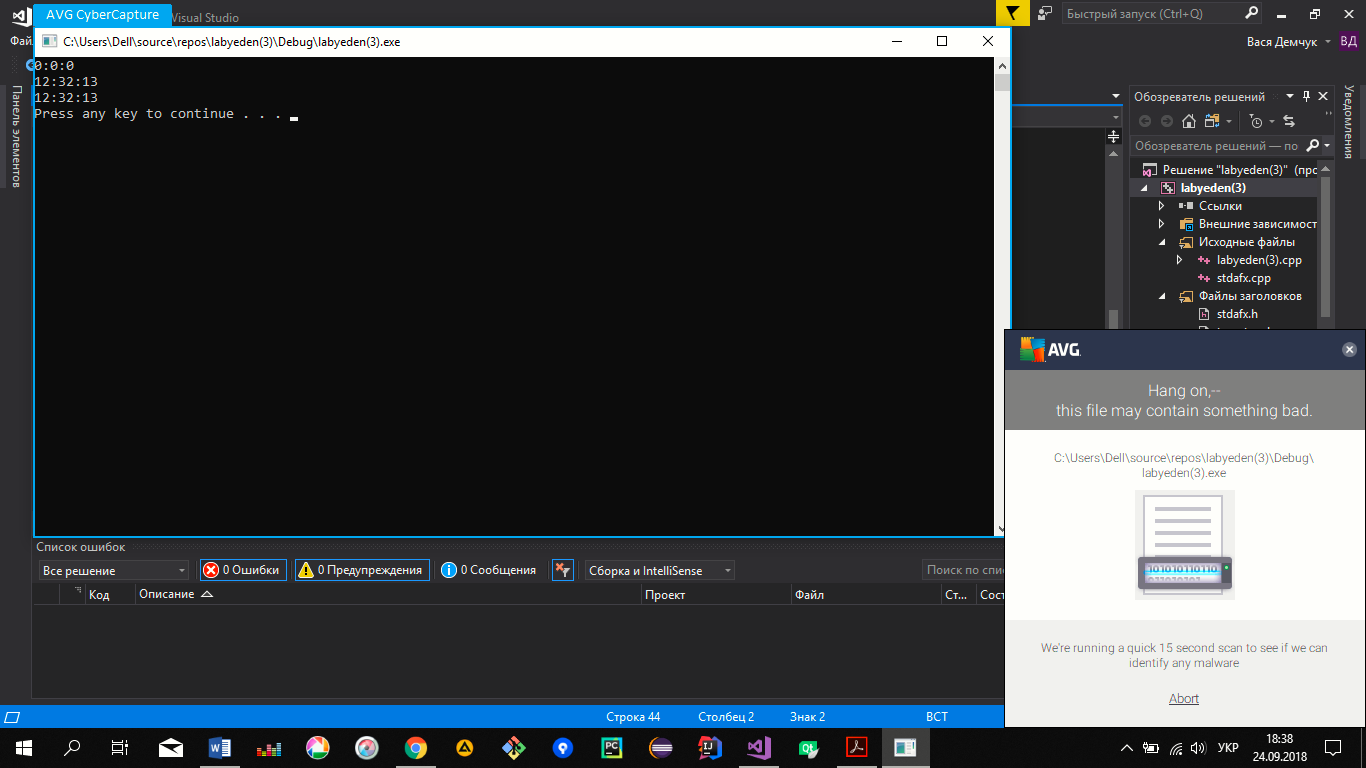
time3.getTime();

system("pause");

return 0;

}

Вивід програми



Висновок: Закріпив базові знання про клас. Навчивсь створювати класи засобами мови С++